

# 2020 자율주행 교육

---

WeGo 위고 주식회사

# 01

---

Darknet & Yolo

## 01 Darknet & Yolo

- Darknet 홈페이지를 참고하여, Darknet 및 Yolo 설치 진행
- <https://pjreddie.com/darknet/>
- git clone <https://github.com/pjreddie.com/darknet.git>
- 위의 명령어를 이용하여, Darknet Framework Download 가능



## 01 Darknet & Yolo

- 이후, wget 명령어 이용하여, 필요한 Weight 다운로드
- wget <https://pjreddie.com/media/files/yolov3.weights>
- 인터넷 상태에 따라 시간 소요



## 01 Darknet & Yolo

- Darknet 내부 구조
- Tree 명령어를 이용하여, 내부 구조를 효율적으로 파악이 가능

```
12 directories, 1083 files
wego@wego-GF63-Thin-10SCXR:~/darknet$ tree -d
.
├── backup
├── cfg
├── data
│   ├── img
│   └── labels
├── examples
├── include
├── obj
├── python
├── results
├── scripts
└── src

12 directories
wego@wego-GF63-Thin-10SCXR:~/darknet$
```

## 01 Darknet & Yolo

- 외부의 Makefile의 경우, Make 파일의 구조를 이해할 시, 쉽게 이해 가능.
- 일반적으로 작성할 일이 잘 없으며, 사용자가 변경 가능한, GPU, OPENCV, OPENMP, CUDNN, DEBUG 등의 사용 유무를 결정하여 빌드할 수 있음  
→ 변경 후, 꼭 make 진행

```
wego@wego-GF63-Thin-
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
GPU=1
CUDNN=0
OPENCV=1
OPENMP=0
DEBUG=0

ARCH= -gencode arch=compute_50,code=sm_50 \
      -gencode arch=compute_52,code=sm_52 \
      -gencode arch=compute_60,code=sm_60 \
      -gencode arch=compute_61,code=sm_61 \
      -gencode arch=compute_70,code=sm_70 \

#      -gencode arch=compute_20,code=[sm_20,sm_21] \ This one is deprecated?

# This is what I use, uncomment if you know your arch and want to specify
# ARCH= -gencode arch=compute_52,code=compute_52

VPATH=./src/./examples
SLIB=libdarknet.so
ALIB=libdarknet.a
```

## 01 Darknet & Yolo

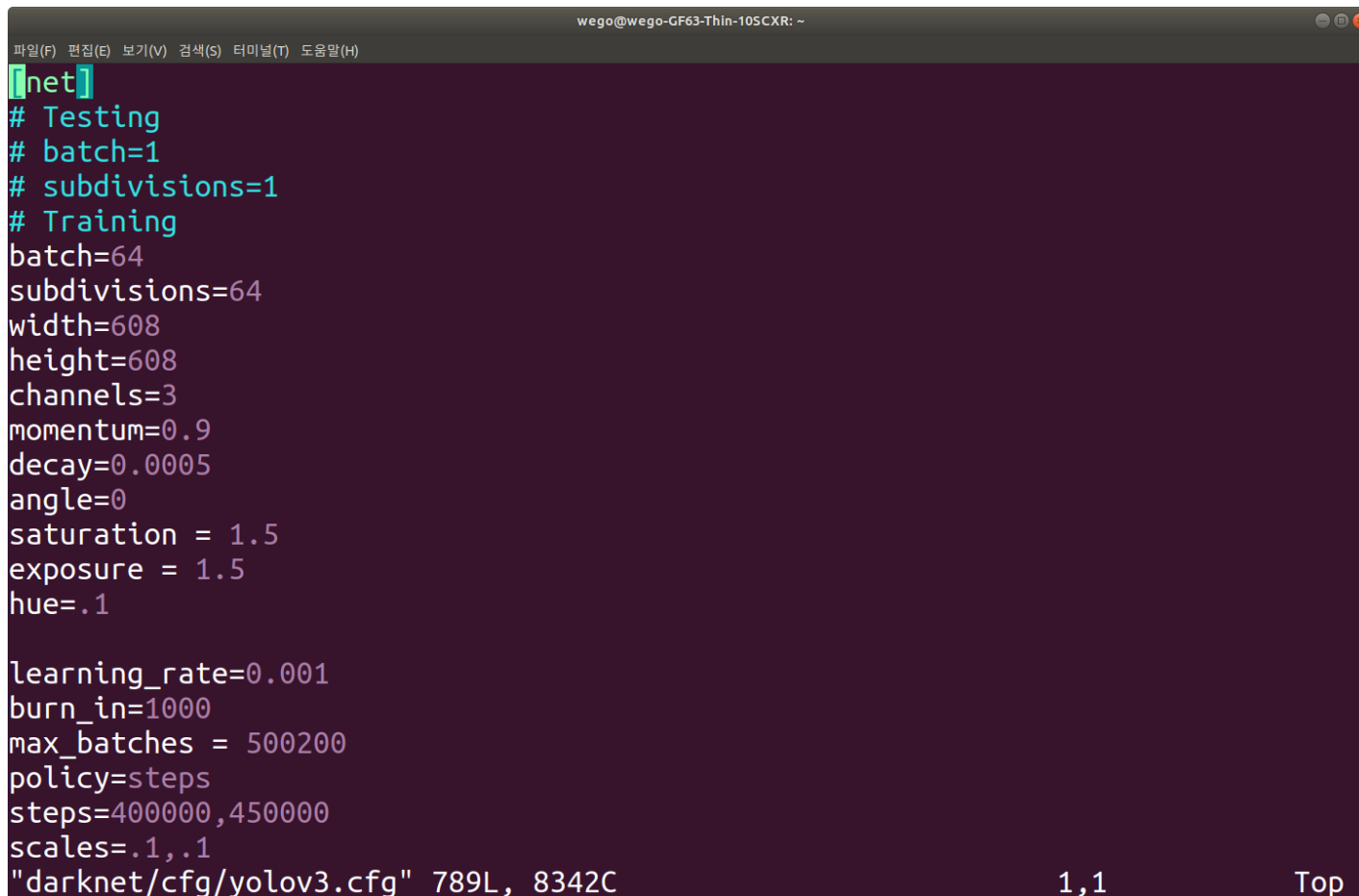
- Backup 폴더의 경우, PC가 특정한 상황에서 전원이 종료되거나 또는 Training 과정에서 문제가 발생하여, 중간부터 새롭게 Training을 진행할 경우, 불러서 사용할 수 있는 Weight값이 저장되는 장소,

```
12 directories, 1083 files
wego@wego-GF63-Thin-10SCXR:~/darknet$ tree -d
.
├── backup
├── cfg
├── data
│   ├── img
│   └── labels
├── examples
├── include
├── obj
├── python
├── results
├── scripts
└── src

12 directories
wego@wego-GF63-Thin-10SCXR:~/darknet$
```

## 01 Darknet & Yolo

- cfg 폴더의 내부의 cfg파일을 살펴보면 유명한 Network(AlexNet, ResNet 등)의 구조가 적혀있으며, 사용하는 YOLOv3의 구조 또한 짜여져 있는 것을 확인할 수 있음.
- 새롭게 네트워크를 구축하거나, Parameter를 변경 시, 적용하여 사용 가능



```
wego@wego-GF63-Thin-10SCXR: ~
[net]
# Testing
# batch=1
# subdivisions=1
# Training
batch=64
subdivisions=64
width=608
height=608
channels=3
momentum=0.9
decay=0.0005
angle=0
saturation = 1.5
exposure = 1.5
hue=.1

learning_rate=0.001
burn_in=1000
max_batches = 500200
policy=steps
steps=400000,450000
scales=.1,.1
"darknet/cfg/yolov3.cfg" 789L, 8342C 1,1 Top
```



## 01 Darknet & Yolo

- cfg폴더 내부의 coco.data를 살펴보면, 아래와 같이 분류가 가능한 class의 개수
- train 및 validation set에 대한 정보, backup 장소 및 class의 이름에 대한 정보가 정리되어 있음.

[illegible]

## 01 Darknet & Yolo

- 유사하게 Data폴더 아래의 coco.names를 살펴보면 위에서 확인하였던 80개의 class에 대한 이름이 정리되어 있음.
- 이 후, Training을 진행할 시, 이를 변경 및 수정하여 Training에 활용 가능

```
wego@wego-GF63-Thin-105CXR: ~  
person  
bicycle  
car  
motorbike  
aeroplane  
bus  
train  
truck  
boat  
traffic light  
fire hydrant  
stop sign  
parking meter  
bench  
bird  
cat  
dog  
horse  
sheep  
cow  
elephant  
"darknet/data/coco.names" 80L, 625C 1,1 Top
```

